Universal Serial Bus
Device Class Definition
for Mass Storage Devices

0.90c Draft Revision
February 2, 1996

Scope of this Revision

The 0.9c release candidate of this definition is intended for industry review.

Contributors

Curtis E. Stevens Phoenix Technologies, Curtis_Stevens@PTLTD.COM
Mark Gianopulos Intel Corporation, Memory Card Division, Mark_P_Gianopulos@ccm.fm.intel.com

Revision History

Rev	Date	Filename	Author	Comments
0.9b	1/15/96	massr1i.doc	Carol Buchmiller	Document Normalized to match other Class Specifications.
0.9c	2/2/96	massr1j	Curtis E. Stevens	Added Company/Address information for Mark Gianopulos. Insterted copyright Changed Revision to C.

Please send comments via electronic mail to usbdevice@fsp008.fm.intel.com

Table of Contents

List of Tables

1. Introduction

The Universal Serial Bus (USB) is a communications architecture that gives a PC the ability to interconnect a variety of devices via a simple four-wire cable. The USB is actually a two-wire serial communication link that runs at 12 megabits (Mbs) per second. USB protocols can configure devices at startup or when they are plugged in at run time. These devices have been broken into five major classes:

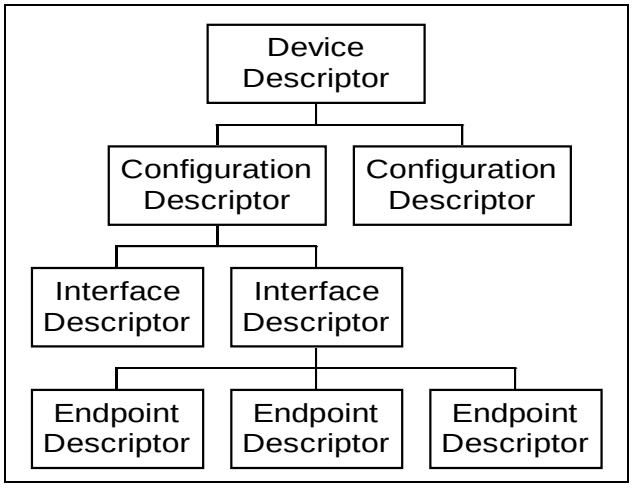
- Monitors
- Communication devices
- Audio
- Human input
- Mass storage

More classes will be identified as the USB matures.

Devices in a given class can be broken into subclasses. These divisions allow software to search the bus and select all of the devices that it can control. Each device can have one or more configurations that define how the device functions. A device that can function in several different ways will have a configuration for each function. A *configuration* is a collection of interfaces. An *interface* specifies which hardware in the device will interact with the USB. Each of these pieces of hardware is called an *endpoint*. Therefore, an interface is a collection of endpoints. The USB Device Class Definitions define the default configurations, interfaces and endpoints that a device in a given class or subclass should provide.

A *descriptor* describes general information about a device, configuration, interface, or endpoint. Figure 1 shows the hierarchical organization of USB descriptors.

Figure 1. Sample Descriptor Organization



1.1 Scope

This document fully describes the Mass Storage Class (MSC) of USB devices. It describes:

- The Mass Storage Class and its subclasses
- Device, configuration, interface, and endpoint descriptors
- The USB standard requests used by Mass Storage devices

This document lists, but does not describe, device-specific command sets used by the General Mass Storage, CD-ROM, and Tape subclasses of Mass Storage device. For a description of the command set used by Solid State devices, see Appendix B, “Solid State SCSI Command Set,” in this document.

1.2 Purpose

The purpose of this document is to provide some common descriptions of the USB configuration, interface, and endpoint descriptors, as well as a communications protocol, for operating system, BIOS, and peripheral designers implementing support for Mass Storage devices. These definitions allow an operating system designer to design a single software package to support a given class or subclass of device. These definitions also provide a framework for designing the peripherals in each class or subclass. Mass Storage devices differ from other USB devices in that they can be used to store and boot an operating system. This means that the system BIOS must be able to control all devices of Mass Storage.

1.3 Related Documents

The Mass Storage Class (MSC) uses the command sets from several existing protocols. These command sets are placed in a USB wrapper and follow USB protocol. The following specifications are referenced by the MSC:

- *Small Computer System Interface - 2* (SCSI-2). ANSI document X3.131.
- *Advanced Technology Attachment Packet Interface (ATAPI) for CD-ROMs*. SFF-8020.
- *Advanced Technology Attachment Packet Interface (ATAPI) for Tape*. QIC-157.
- *Universal Serial Bus Specification*, 1.0 final draft revision (also referred to as the *USB Specification*). In particular, see Chapter 9, “USB Device Framework.”
- *Enhanced Disk Drive (EDD)*, Revision 1.1. Phoenix Technologies LTD.
- *El Torito CD-ROM Boot Specification.* Phoenix Technologies LTD and IBM Corporation.
- *USB Device Class Definition for Audio Devices.*

1.4 Terms and Abbreviations

CHS	Cylinders, heads, and sectors. A convention for device addressing used by older operating systems.
CPU data	Data that is normally used by the microprocessor. A conventional CD-ROM can contain programs in the form of CPU data and music in the form of audio data.
MSC	Mass Storage Class.
Streaming data	Data received sequentially and stored linearly on the media by a Tape device.

2. Management Overview

This document defines the following distinct subclasses of USB Mass Storage Class devices:

- **General Mass Storage subclass**. Mass Storage devices are normally used in a random access fashion. The General Mass Storage subclass includes storage devices such as the following:
  - Conventional floppy
  - Magneto-optical
  - Zip (floptical)
  - Syquest
  - Hard drives

  The command set for this subclass does not distinguish between devices with fixed and removable media. Commands for ejecting and unlocking fixed media will simply fail.
- **CD-ROM**. CD-ROM drives are unique because, in addition to conventional data transfers, they are capable of playing audio and video. This means it is possible for a CD-ROM drive to require two separate isochronous channels.
- **Tape**. Tape drives are unique because they require *streaming* data. This means that the device takes a sequential stream of data and stores it linearly on the media. The tape subclass is oriented toward providing the drive with information on time. If data is not available or data corruption occurs, the drive must perform a time-consuming retry procedure.
- **Solid State**. Solid State devices are unique because they have no moving parts. These devices generally read very quickly, but require an erase operation before a write can occur.

3. Functional Characteristics

This section describes the functional characteristics of Mass Storage devices, including:

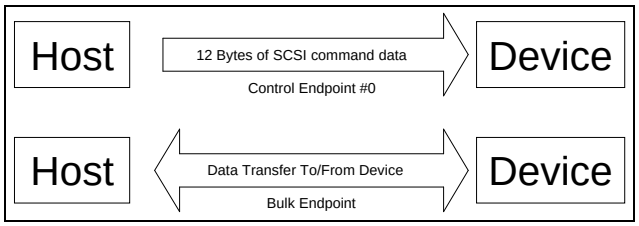
- The operational model
- Interfaces
- Characteristics of each subclass: General Mass Storage, CD-ROM, Tape, and Solid State

3.1 Operational Model

Devices of Mass Storage have two different types of commands: those that transfer data, and those that control the mechanism. Both of these command types follow the same procedure for accomplishing their tasks: the host issues a 12-byte command and the device responds by accepting or returning data.

All Mass Storage devices use a 12-byte SCSI command set enclosed in a USB wrapper. This USB wrapper takes the following form:

Figure 2. A Typical Command Sequence



Recoverable errors are only reported in response to a request from the host. Non-recoverable errors generate a STALL on the appropriate endpoint.

The Host sends commands to the device via the Control endpoint (endpoint 0). The device can respond by transferring data—either status information or information transferred from the media—on a bulk endpoint. If the device has some abnormal status to report after the data transfer is complete, it generates a USB STALL on the appropriate bulk endpoint. If the command issued by the host does not require data transfer, none is generated; in case of abnormal status on this type of command, the device stalls the Control Endpoint. The host must then query the device via a **Request Sense** command to clear the STALL condition. **Caution: A STALL should only be generated for a hard device error. Sense conditions that do not indicate data corruption should not generate a STALL.**

Devices can collect data on recoverable errors and report the information in response to a request from the host. In the unlikely event that a Mass Storage device becomes non-operational, the host can issue a SEND DIAGNOSTIC command on the Control endpoint. USB devices use the SEND DIAGNOSTIC command to reset the device and perform a self test**. All USB Mass Storage Devices perform a hard RESET in response to SEND DIAGNOSTIC.**

3.2 Interfaces

Mass Storage devices need to support only a single interface, the Data interface. This interface is enabled when the default configuration is selected. It allows data to be transferred to or from the device as well as providing an Interrupt to signal unscheduled events such as media change. This interface provides the following endpoints:

- Bulk Input endpoint
- Bulk Output endpoint
- Interrupt endpoint

CD-ROM drives can support both audio and video isochronous data transfers as well as bulk transfers. For this reason, a CD-ROM device can have up to three interfaces: Data, Audio and Video. Because video transfers can consume a considerable portion of the USB bandwidth, this interface is optional.

3.3 Functional Characteristics of Mass Storage Device Subclasses

The rest of this section describes functional characteristics of each subclass of Mass Storage device.

3.3.1 General Mass Storage Subclass

The General Mass Storage subclass includes devices other than CD-ROM, tape, or solid state. Typically, this category includes random-access devices that are both readable and writable. This device subclass is capable only of transferring data, so it does not require any isochronous endpoints. The General Mass Storage subclass is designed around removable media devices such as floppy, MO, ZIP, and Syquest technology. Devices with non-removable media such as hard drives also reside in this class. A normal hard drive is simply a Mass Storage device in which the media is permanently locked.

General Mass Storage devices are random-access, block/sector oriented. These have the ability only to store and retrieve CPU data. The interface to these devices follows the Direct Access Storage Device protocol of the SCSI-2 standard. The media on USB devices is accessed using logical blocks in the same manner as SCSI-2 devices.

For backward compatibility, some of these devices accept media from older operating systems that use cylinders, heads and sectors (CHS) for device addressing. This means the device must be capable of sensing and reporting the geometry of the media. A good example of this is a floppy that was formatted under MS-DOS 6.22. If this floppy is inserted into a USB floppy drive, the drive must be able to report to the operating system the type of floppy inserted (360k, 720k, 1.2MB, 1.4MB, 2.88MB, etc.). This information gives the BIOS, or OS, a way of converting the CHS addressing information into logical blocks.

3.3.2 CD-ROM Subclass

A CD-ROM drive is a read-only device that can transfer binary data, sound, and possibly video. CD-ROM is by far the most complex of the Mass Storage devices, because it has independent interrupt, CPU data, audio, and video capabilities. The video endpoint is included here mainly for future use.

CD-ROM drives are block/sector oriented devices. CD-ROMs have the ability to store three different kinds of information: CPU data, audio, and video. Sector sizes vary based on the type of information stored: CPU data is normally stored in 2048-byte sectors, while audio and video can vary from 2000 bytes to over 3000 bytes. **Read** commands access data, while a **Play** command plays back audio. Audio and video information can be read as data by a conventional **Read** command.

Although a CD-ROM drive is a read-only device, the data interface allows for bidirectional data transfers. It is obvious that data must be transferred from a CD-ROM drive; however, CD-ROM drives may also need to receive data because commands such as **Mode Select** need parameter information.

The CD-ROM subclass supports an interrupt; however, the only function of this interrupt is to signal a media change.

3.3.3 Tape Subclass

A Tape Storage device is any device that has a removable media and requires streaming data. This device subclass is capable only of transferring data, so it does not require any isochronous endpoints. The interface to these devices follows the *Advanced Technology Attachment Packet Interface* (ATAPI) for Tape of the QIC-157 standard.

Tape Storage devices are pseudo-random access block-oriented drives. The term pseudo-random is used because these devices have potentially very long seek times, measured in seconds and sometimes minutes. Tape Storage devices require a steady stream of data. If there is an interruption in the data, the device must seek to a new location and restart the read or write operation.

3.3.4 Solid State Subclass

A Solid State device is any linear memory device that has no moving parts and requires memory regions to be erased before individual bytes of data can be written. This device subclass is capable only of transferring data, so it does not require any isochronous endpoints. The interface to these devices borrows from the *Direct Access Storage and Optical Memory Device* protocol of the SCSI-2 standard.

Solid State devices are random-access byte-oriented devices. By definition, seek time on these devices is 0. Read operations operate at the full speed the USB can provide. Depending on the device type, write operations are typically preceded by an erase operation. For solid state devices such as flash memory, if a write operation will change any bit from a programmed state to an erased state, the entire block must first be erased before the write operation can occur.

Unscheduled events, such as media change, are generated via the Interrupt endpoint. A single word packet identifies the event that caused the interrupt.

Although the Solid State device will utilize many of the actual packet definitions from the SCSI standard, it will NOT use most other features of the normal SCSI Protocol. Thus there are no phases, no messages, no shareable bus (only one host), and no SCSI hardware. This specification will make use of many of the Standard SCSI Command Descriptor Block definitions and commands, but some of the commands that would normally be supported by a SCSI device will not be supported for various reasons.

Some of the major differences from the SCSI Standard include:

- The **Inquiry** command has been modified in that some of the vendor unique fields for Solid State devices have been redefined.
- Unused fields have been marked as Reserved.

4. Standard Descriptors

Mass Storage Class devices support the following standard USB descriptors:

- **Device**. Each drive has one device descriptor.
- **Configuration**. Each device has one default configuration descriptor, which supports at least one interface.
- **Interface**. General Mass Storage, Tape, and Solid State devices have a single data interface. CD-ROM drives have three possible interfaces: CPU data, audio, and video.
- **Endpoint**. A Mass Storage device supports the following endpoints:
    - Control endpoint. USB required endpoint.
    - Interrupt endpoint. Signals media change for all Mass Storage device subclasses. In the case of the Solid State subclass, also signals Unscheduled Event Notification.
    - Bulk Input endpoint. CPU data transfers.
    - Bulk Output endpoint. CPU data transfers.

Mass Storage Devices have no class-specific descriptors.

The rest of this section describes the standard USB device, configuration, interface, and endpoint descriptors for Mass Storage devices. For information about other standard descriptors, see Chapter 9, “USB Device Framework,” of the *USB Specification*.

4.1 Device Descriptor

There is only one Device Descriptor for each USB device. This descriptor contains the definitions of the device class and the device subclass, among other things.

Table 1 - Device Descriptor

Offset	Field	Size	Value	Description
0	bLength	Byte	12h	Size of this descriptor in bytes.
1	bDescriptorType	Byte	01h	DEVICE descriptor type.
2	bcdUSB	Word	0100h	USB Specification Release Number in BCD.
4	bDeviceClass	Byte	01h	Base class for Mass Storage devices.
5	bDeviceSubClass	Byte	01h	The subclass code of Mass Storage device: 01 General Mass Storage 02 CD-ROM 03 Tape 04 Solid State
6	bDeviceProtocol	Byte	00h	Mass Storage devices do not use class-specific protocols.
7	wMaxPacketSize0	Byte	40h	Maximum packet size for endpoint 0.
6	idVendor	Word	????h	Vendor ID (assigned by the USB).
8	idProduct	Word	????h	Product ID (assigned by the manufacturer).
10	bcdDevice	Word	????h	Device release number in BCD.
14	iManufacturer	Byte	??h	Index of string descriptor describing manufacturer.
15	iProduct	Byte	??h	Index of string descriptor describing this product.
16	iSerialNumber	Byte	??h	Index of string descriptor describing the device's serial number.
17	bNumConfigurations	Byte	01h	Number of possible configurations. There must be at least one default configuration.

4.2 Configuration Descriptor

A Mass Storage device has one default configuration descriptor. This descriptor has one interface, called the Data interface, which has four endpoints: Control, Bulk Input, Bulk Output, and Interrupt. The **bLength** field of the configuration descriptor specifies the length of this descriptor, not the length of the data returned by the **Get Configuration** request.

Table 2 - Configuration Descriptor

Offset	Field	Size	Value	Description
0	bLength	Byte	09h	Size of this descriptor in bytes.
1	bDescriptorType	Byte	02h	CONFIGURATON descriptor type.
2	bTotalLength	Word	????h	Number of bytes in this configuration. This includes the configuration descriptor plus all of the interface and endpoint descriptors.
4	bNumInterfaces	Byte	??h	For General Mass Storage, Tape, and Solid State devices, this configuration has only one interface. For CD-ROM drives, this configuration has at least two and possibly three interfaces.
5	bConfigurationValue	Byte	01h	Value to write to the DCR to select this configuration.
6	iConfiguration	Byte	??h	Index of string descriptor describing this configuration.
7	bmAttributes	Byte	40h	Configuration characteristics: Bit Description 7 Bus-powered 6 Self-powered 5 Remote wakeup 4-0 Reserved, set to 0 Mass Storage devices are usually self-powered, but may be both self-powered and bus-powered.
8	MaxPower	Byte	??h	Maximum power consumption of this configuration. Units are mA/2.

4.3 Interface Descriptors

All Mass Storage devices support a Data interface that provides the means for data to be transferred to or from the device. This interface also supports an interrupt for signaling unexpected events such as a media change.

A CD-ROM device supports at least one additional interface for audio, and may support a second additional interface for video. When the default configuration is selected, the audio interface is disabled. This allows minimal control of CD-ROM drives by limited software such as a system BIOS.

The definition of the audio and video interfaces is beyond the scope of this document. For details about audio interfaces, see the *USB Device Class Definition for Audio Devices*. For details about video interfaces, see the *USB Device Class Definition for Video Devices* (to be written).

Table 3 - Data Interface Descriptor

Offset	Field	Size	Value	Description
0	bLength	Byte	09h	Size of this descriptor in bytes.
1	bDescriptorType	Byte	04h	INTERFACE descriptor type.
2	bInterfaceNumber	Byte	00h	Zero-based value identifying the number of this interface. General Mass Storage, Tape, and Solid State devices use only one interface with the value 00h. CD-ROM drives use two and possibly three interfaces: 00h Data interface 01h Audio interface 02h Audio-With-Video interface
3	bAlternateSetting	Byte	??h	Value used to select an alternate interface.
4	bNumEndpoints	Byte	03h	Number of endpoints used by this descriptor.
5	bInterfaceClass	Byte	00h	This interface is specific to Mass Storage devices.
6	iInterfaceSubClass	Byte	00h	This interface is specific to Mass Storage devices.
7	bInterfaceProtocol	Byte	00h	This interface does not require a class-specific protocol.
8	iInterface	Byte	??h	Index to string describing this interface.

4.4 Endpoint Descriptors

A Mass Storage device supports a minimum of four endpoints.

A CD-ROM device supports these four endpoints plus those required for audio and video, if supported.

4.4.1 Control Endpoint

Every USB device defines a Control endpoint. This endpoint is used for both global USB commands and device-specific commands. All USB Mass Storage devices respond to SCSI commands issued to this endpoint. This is a default endpoint which does not require a descriptor. The Device descriptor specifies the number of bytes in the Control endpoint.

4.4.2 Bulk Input Endpoint

Read commands use a Bulk endpoint to transfer data from the device or to return status information about the device.

Table 4 - Bulk Input Endpoint Descriptor

Offset	Field	Size	Value	Description
0	bLength	Byte	09h	Size of this descriptor in bytes.
1	bDescriptorType	Byte	05h	ENDPOINT descriptor type.
2	bEndpointAddress	Byte	80h	The address of this endpoint on the USB device. This address is an endpoint number between 0 and 15. Bit 0..3 Endpoint number Bit 4..6 Reserved, must be 0 Bit 7 0 = Out, 1 = In
3	bmAttributes	Byte	02h	This is a Bulk endpoint.
4	wMaxPacketSize	Word	0064h	Maximum data transfer size.
6	wSampleSize	Word	0000h	Does not apply to Bulk endpoints.
8	bInterval	Byte	00h	Does not apply to Bulk endpoints.

4.4.3 Bulk Output Endpoint

The Bulk Output endpoint is used for transferring data from the host through the device to the media. In the case of CD-ROM devices, some ATAPI commands require additional parameter information. This endpoint allows that data to be transferred.

Table 5 - Bulk Output Endpoint Descriptor

Offset	Field	Size	Value	Description
0	bLength	Byte	09h	Size of this descriptor in bytes.
1	bDescriptorType	Byte	05h	ENDPOINT descriptor type.
2	bEndpointAddress	Byte	01h	The address of this endpoint on the USB device. This address is an endpoint number between 0 and 15. Bit 0..3 Endpoint number Bit 4..6 Reserved, must be 0 Bit 7 0 = Out, 1 = In
3	bmAttributes	Byte	02h	This is a Bulk endpoint.
4	wMaxPacketSize	Word	0064h	Maximum data transfer size.
6	wSampleSize	Word	0000h	Does not apply to Bulk endpoints.
8	bInterval	Byte	00h	Does not apply to Bulk endpoints.

4.4.4 Interrupt Endpoint

The Interrupt endpoint signals a media change for General Mass Storage, CD-ROM, and Tape devices, or an unscheduled event such as media change or erase/write completion for a Solid State device. It should respond within one second of the actual change event (except for CD-ROM devices, which should respond within two seconds).

When an interrupt is generated, the device will return 16 bits of status information. The high order 8 bits are the device-specific ASC code describing the condition (SCSI-2 for General Mass Storage, SFF-8020 for CD-ROM, QIC-157 for Tape and Solid State). The low order 8 bits are the device-specific ASCQ code describing the condition.

Table 6 - Interrupt Endpoint Descriptor

Offset	Field	Size	Value	Description
0	bLength	Byte	09h	Size of this descriptor in bytes.
1	bDescriptorType	Byte	05h	ENDPOINT descriptor type.
2	bEndpointAddress	Byte	02h	The address of this endpoint on the USB device. This address is an Endpoint number between 0 and 15. Bit 0..3 - Endpoint number Bit 4..6 - Reserved, must be 0 Bit 7 - 0 = Out, 1 = In
3	bmAttributes	Byte	03h	This is an Interrupt endpoint.
4	wMaxPacketSize	Word	0002h	This returns 16 bits (1 word) of status data.
6	wSampleSize	Word	0000h	Does not apply to Interrupt.
8	bInterval	Byte	??h	General Mass Storage, CD-ROM, Tape subclasses: This field does not apply to Interrupt. Its value is 00h. Solid State subclass: Interval for polling the endpoint for data transfers. Expressed in milliseconds in the range 1 to 255.

5. Requests

A Mass Storage device can respond to two different types of requests:

- Standard USB device requests, which perform general functions for supporting the bus and bus-related functions.
- Class specific requests, which cause the device to transfer data to or from the device or, in the case of CD-ROM devices, cause the CD-ROM drive to either transfer data or play audio.

5.1 Standard Requests

A Mass Storage device supports all of the standard device requests described in Chapter 9, “Device Framework,” of the *USB Specification*:

- Clear Stall
- Get Configuration
- Get Descriptor
- Get Interface
- Get Status
- Set Address
- Set Configuration
- Set Descriptor
- Set Idle
- Set Interface
- Set Remote Wakeup

5.2 Class-Specific Requests

A Mass Storage device supports one class-specific request on the Control endpoint (endpoint 0):

- **Accept Device-Specific Command** (ADSC)

The physical request number is 0. When this request is issued, the device-specific command will be sent during the USB data phase.

Each Mass Storage device subclass must support a minimum set of device-specific commands. For a table of device-specific commands, see Appendix A, “Required Device-Specific Commands by Subclass,” in this docu*ment.

Appendix A Required Device-Specific Commands by Subclass

This appendix lists the minimum set of commands that must be implemented for each subclass of Mass Storage device. A vendor can implement additional commands and supply a driver to take advantage of them. For actual command formats, see the following documents:

- General Mass Storage - ANSI X3.131 - *Small Computer Systems Interface - 2*
- CD-ROM - SFF-8020i, *ATA Packet Interface for CD-ROMs*
- Tape - QIC-157, *ATA Packet Interface for Tape*
- Solid State - QIC-157, *ATA Packet Interface for Tape,* with modifications described in Appendix B, “Solid State SCSI Command Set,”in this document.

Table 7 - Required Commands By Subclass

Command	Description	General Mass Storage	CD-ROM	Tape	Solid State
Copy	Duplicate a block of data on the same media				√
Erase (12)	Erase the media.			√	√
Format	Format unformatted media.	√			
Inquiry	Get media information.	√	√	√	√
Load/Unload	Request a removable-media device to load or unload its media.	√	√	√	
Locate	Reposition the tape.			√	
Mode Select	Allow the host to set parameters in a peripheral. **Mode Sense** should be issued prior to a **Mode Select**.	√	√	√	
Mode Sense	Report parameters to the host. Backward compatibility of floppy drives requires support for the **Mode Sense** command, **Flexible Disk** page.	√	√	√	
Pause/ Resume	Pause or resume a CD-ROM.		√		
Play CD	Begin playing an audio CD.		√		
Prevent/ Allow Medium Removal	Prevent or allow the removal of media from a removable media device.	√	√	√	
Read (10)	Transfer binary data from the media to the host.	√	√		
Read (12)	Transfer binary data from the media to the host.			√	√

Read CD-ROM Capacity	Return the amount of space used on the CD.		√		
Read TOC	Return the table of contents from a CD.		√		
Request Sense	Transfer status sense data to the host.	√	√	√	√
Scan	Request a fast forward or fast reverse on a CD-ROM drive.		√		
Seek (10)	Seek the device to a specified address.	√	√		
Send Diagnostic	Perform a hard reset and execute diagnostics.	√	√	√	√
Test Unit Ready	Request the device to report if it is ready.	√	√	√	√
Verify	Compare data on the media with supplied data.	√			
Write (10)	Transfer binary from the host to the media.	√			
Write (12)	Transfer binary from the host to the media.			√	√

Appendix B Solid State SCSI Command Set

The Solid State device-specific commands use the 12-byte command packet of the SCSI - 2 specification with one addition: the **Write Buffer** command. The following outlines the Solid State packet structure. All reserved bits are zero (0).

Table 8 - Packet Commands Supported by Flash Devices

Command Description	Operation Code	Type	Media Access
Copy	18h	O	√
Erase (12)	ACh	M	√
Inquiry	12h	M	
Read (12)	A8h	M	√
Request Sense	03h	M	
Write Buffer	3Bh	O	
Write (12)	AAh	M	√

M = command implementation is mandatory.
O = command implementation is optional.

B.1 Command Set Documentation

This section describes each device-specific command supported by the Solid State subclass of Mass Storage device.

B.1.1 Copy Command

The **Copy** command provides a means to copy data from one area of the medium to another without transferring the data over the USB wire. **Parameter List Length** is the length in bytes of the parameter list table, which contains a Solid State-specific segment descriptor that describes the address and length of the copy.

Table 9 - Copy Command

Byte	D7	D6	D5	D4	D3	D2	D1	D0
0	Operation code (18h)							
1	Reserved			Reserved				Reserved
2 - 4	(MSB)	Parameter List Length(10h)						(LSB)
5 - 11	Reserved							

The **Copy** parameter list table begins with a 4-byte header that contains the **Copy** function code and reserved fields not used by Solid State devices. Following the header is the Solid State segment descriptor, which contains three fields: **Source Address**, **Destination Address**, and **Transfer Length**. This segment descriptor is identified by the vendor-unique **Copy** function code 10h.

Table 10 - Copy Parameter List

Byte	D7	D6	D5	D4	D3	D2	D1	D0
0	Copy function code (10h)					Reserved		
1 - 3	Reserved							
4 - 7	(MSB)	Source Address						(LSB)
8 - 11	(MSB)	Destination Address						(LSB)
12 - 15	(MSB)	Transfer Length						(LSB)

B.1.2 Erase Command

The **Erase** command causes part or all of the medium to be erased. Address is the physical erase block address, and should point to an address in the block to be erased. **Count** is the number of consecutive blocks to be erased. Setting the erase all (ERA) bit to 1 indicates that all blocks on the medium should be erased. In this case, **Address** and **Count** are ignored.

Table 11 - Erase Command

Byte	D7	D6	D5	D4	D3	D2	D1	D0
0	Operation code (ACh)							
1	Reserved					ERA	Reserved	Reserved
2 - 5	(MSB)	Address						(LSB)
6 - 9	(MSB)	Count						(LSB)
10 - 11	Reserved							

B.1.3 Inquiry Command

The **Inquiry** command requests that information regarding parameters of the device be sent to the host.

Table 12 - Inquiry Command

Byte	D7	D6	D5	D4	D3	D2	D1	D0
0	Operation code (12h)							
1	Reserved			Reserved			Reserved	
2 - 3	Reserved							
4	Allocation Length							
5 - 11	Reserved							

Refer to the SCSI-2 specification for a detailed explanation of fields 0 - 35.

Table 13 - Inquiry Data Format

Byte	D7	D6	D5	D4	D3	D2	D1	D0
0	Reserved			Reserved				
1	RMB	Reserved						
2 - 3	Reserved							
4	Additional Length (51)							
5 - 7	Reserved							
8 - 15	Vendor Identification							
16 - 31	Product Identification							
32 - 35	Product Revision Level							
36 - 37	JEDEC Identifier							
38 - 41	Media Size							
42 - 45	Erase Block Size							
46 - 47	Number of Erase Blocks per Device							
48 - 49	Number of Devices							
50	Device Size							
51	Reserved							MLC
52 - 55	Reserved							

B.1.4 Read Command

The **Read** command requests that the device transfer one or more bytes of data to the host, starting with the byte at **Address**.

Table 14 - Read Command

Byte	D7	D6	D5	D4	D3	D2	D1	D0
0	Operation code (A8h)							
1	Reserved			Reserved	Reserved	Reserved		Reserved
2 - 5	(MSB)	Address						(LSB)
6 - 9	(MSB)	Transfer Length						(LSB)
10 - 11	Reserved							

B.1.5 Request Sense Command

The **Request Sense** command requests that the target transfer sense data to the initiator. This command is used to report errors to the host and to clear STALL conditions. The **Request Sense** data format is defined in the SCSI - 2 specification.

Table 15 - Request Sense Command

Byte	D7	D6	D5	D4	D3	D2	D1	D0
0	Operation code (03h)							
1	Reserved			Reserved				
2 - 3	Reserved							
4	Allocation Length							
5 - 11	Reserved							

B.1.6 Write Buffer Command

The **Write Buffer** command is used to update or replace the software components on the device.

Table 16 - Write Buffer Command

Byte	D7	D6	D5	D4	D3	D2	D1	D0
0	Operation code (3Bh)							
1	Reserved					Mode (101b)		
2	Buffer ID (0)							
3 - 5	Reserved							
6 - 8	(MSB)	File Size in Bytes						(LSB)
9 - 11	Reserved							

B.1.7 Write Command

The **Write** command requests that the device write the data that is transferred from the host, starting at Address.

Table 17 - Write Command

Byte	D7	D6	D5	D4	D3	D2	D1	D0
0	Operation code (AAh)							
1	Reserved			Reserved	Reserved	Reserved	Reserved	Reserved
2 - 5	(MSB)	Address						(LSB)
6 - 9	(MSB)	Transfer Length						(LSB)
10 - 11	Reserved							

B.2 Example of a Host/Device Message Exchange

The following steps illustrate a typical exchange of messages between the host and a Mass Storage device:

1. The user inserts removable media into the device.

2. The device initiates insertion notification via the Interrupt endpoint. A data packet is sent indicating that media has been inserted (ASC/ASCQ 28/00).

3. The host initiates an **Inquiry** command. If the device is unable to recognize the media, it stalls the endpoint. The host issues a **Request Sense** command and receives an incompatible medium error (ASC/ASCQ 30/00). If the device recognizes the media, it returns information about the size and characteristics of the media.

4. Assuming the device has recognized the media and returned information about it, the host initiates a **Write Buffer** command to update the device software. The device updates the microcode and notifies the host at completion.

5. The host may now issue the **Inquiry** command and begin sending read/write/erase commands.

B.3 Additional Solid State ASC/ASCQ Codes

Two additional codes were added to free host software from having to poll the device for media ready status on **Write** and **Erase** commands. These codes are posted on the Interrupt endpoint after the device has completed the requested function on the media. For example, when the host issues the **Erase** command, the device will accept the command and begin erasing the media. The host is now free to perform other processing while the media is being erased. The Command Complete - Erase (F1/01) code is posted when the device has completed the erase task.

Table 18 - New ASC/ASCQ Codes

ASC	ASCQ	Description
F1	00	Command Complete - Write
F1	01	Command Complete - Erase